1 Scope

The MLX90614 sensors have a build in emissivity compensation function which allows the customer to change the emissivity coefficient without the need of recalibration. The aim of this document is to describe the procedure for determining the new Emissivity (E) coefficient and the necessary calculation in order to upload this newly calculated coefficient into the device EEPROM.

2 Glossary of terms

- To – **O**bject **T**emperature
- Ta – **A**mbient **T**emperature
- BB – **B**lack **B**ody
- T_{BB} – **T**emperature of **BB**
- SA – SMBus **S**lave **A**ddress
- PEC – SMBus **P**acket **E**rror **C**heck

3 When the emissivity needs to be changed

The emissivity is the coefficient which shows how well the object emits IR radiation compared to a theoretical perfect black body emitter. This radiation is used by MLX90614 in order to calculate the temperature of the object. During manufacturing, the MLX90614 is calibrated in front of a black body with an emissivity= 99,9% which we consider as E=1.
Different materials have different emissivity, so in order to measure temperature correctly this should be taken into account by uploading new emissivity coefficient into MLX90614 EEPROM.
So as general rule of thumb compensation is only necessary when the IR radiation from the object is reduced in some of way such as:

- either due to a low emissivity of the object
- either because some IR transparent material (the transparency is always less than 1) is put in front of the MLX90614

4 Determination of the new E

In order to determine what emissivity coefficient we should uploaded into the EEPROM of the MLX90614, the following measurement should be done (for both cases when transparent material and different emissivity surface is used). Please make sure that before starting the procedure the emissivity inside the device is E=1.

4.1 In case a transparent material is put in front of the MLX90614

This is somewhat the easiest case because we can use MLX90614 in order to determine the original temperature of the object (assuming that E=1) before a transparent sheet of material is put between the object and the MLX90614.
The procedure is as follows:

1. Heat up the object to some temperature different from the room temperature. Let us assume $T_O = 60°C$ (please note that this temperature must be kept stable). The temperature difference between the object and the sensor should be at least 30ºC.
2. Run a measurement with MLX90614 and write down the readings as $T_{REAL} = 60°C$ and $T_{A\_REAL} = 25°C$ (those temperatures are just for example)
3. Place the transparent material in front of the MLX90614 (***NOTE: in order to have correct measurements, the temperature of the transparent sheet MUST be the same as the***

ambient temperature of the MLX90614. Otherwise the MLX90614 will "see" the temperature of the sheet as well and this will introduce an error)

4. Run the measurement with MLX90614 and write down the readings as $T_{NEW} = 50°C$ and $T_{A\_NEW} = 25°C$ (in general ambient temperature should be the same)
5. Calculate the new emissivity coefficient using the formulae:

$$E = \frac{T^4_{O\_NEW} - T^4_{A\_NEW}}{T^4_{O\_REAL} - T^4_{A\_REAL}}$$, please note that the temperatures are in Kelvin

$$E = \frac{(50+273{,}15)^4 - (25+273{,}15)^4}{(60+273{,}15)^4 - (25+273{,}15)^4} = \frac{10{,}9047.10^9 - 7{,}9020.10^9}{12{,}3185.10^9 - 7{,}9020.10^9} = 0{,}6799$$

4.2 In case the object has an emissivity different from 1

We need to know the real temperature of the object in order to calculate the new value of the emissivity. Basically there are two methods to determine the real temperature:

1. Using precise contact thermometer to measure the temperature
2. Paint part of the object (the E of the paint should be 1), so you can measure this painted spot with the MLX90614 (please note that the spot must be bigger than the FOV spot of the MLX90614)

Then the procedure is as follows:

1. Heat up the object to some temperature different from the room temperature. Let us assume $T_O = 60°C$ (please note that this temperature must be kept stable)
2. Check the actual temperature using one of the above methods, Let us assume $T_{REAL} = 60°C$ and record the ambient temperature measured by the MLX90614: $T_{A\_REAL} = 25°C$ (these temperatures are just for example)
3. Measure the temperature of the object with the intended surface with the MLX90614 and write down the readings as $T_{NEW} = 40°C$ and $T_{A\_NEW} = 25°C$ (in general the ambient temperature should be the same)
4. Calculate the new emissivity coefficient:

$$E = \frac{T^4_{O\_NEW} - T^4_{A\_NEW}}{T^4_{O\_REAL} - T^4_{A\_REAL}}$$, please note that the temperatures are in Kelvin (add 273,15 to ˚C)

$$E = \frac{(40+273{,}15)^4 - (25+273{,}15)^4}{(60+273{,}15)^4 - (25+273{,}15)^4} = \frac{9{,}6163.10^9 - 7{,}9020.10^9}{12{,}3185.10^9 - 7{,}9020.10^9} = 0{,}3882$$

5 Calculation of the value to be uploaded in MLX90614 EEPROM

Once we calculated the new E, the new values to be loaded in the MLX90614 EEPROM must be calculated.
Please note that the procedure for changing the emissivity setting depends on the device type

5.1 Procedure for MLX90614xAx and MLX90614xBx type devices

This is the simplest procedure as only the cell containing the emissivity information in EEPROM must be changed: this is address 0x04.
We already calculated the new emissivity to be E= 0.6799 or E= 0.3882 (in our examples)
E=1 corresponds to EEPROM content = 0xFFFF. The new values to be stored into EEPROM must be calculated using the expression:

$$0x04d = 2^{16}.E_{NEW} - 1 = 65536X\,0{,}6799 - 1 = 44556{,}926$$, which is rounded to 44557

$$0x04d = 2^{16}.E_{NEW} - 1 = 65536X\,0{,}3882 - 1 = 25440{,}075$$, which is rounded to 25440

This value must be converted from decimal to hexadecimal to get the value to upload in the EEPROM:

$$0x04 = DEC2HEX(44557) = 0xAE0D$$
$$0x04 = DEC2HEX(25440) = 0x6360$$

5.2 Procedure for MLX90614xCx type devices

In this case 2 EEPROM cell values have to be changed: 0x04 and 0x0F.
The calculation of the new content of memory cell 0x04 is exactly as in case 5.1, i.e.:

$$0x04 = DEC2HEX(44557) = 0xAE0D$$
$$0x04 = DEC2HEX(25440) = 0x6360$$

To calculate the new value for cell 0x0F the method is:
As a first step, it is necessary to read the old content of the EEPROM addresses 0x04 and 0x0F and make the following calculation:

$$0x0F_{NEW} = DEC2HEX\left(\frac{HEX2DEC(0x04_{OLD})}{E_{NEW}} HEX2DEC(0x0F_{OLD})\right)$$

If for example 0x04=0xFFFF and 0x0F=0x099A we get:

$$0x0F_{NEW} = DEC2HEX\left(\frac{HEX2DEC(FFFF)}{44557} HEX2DEC(099A)\right)$$

or:

$$0x0F_{NEW} = DEC2HEX\left(\frac{65535}{44557}.2458\right) = DEC2HEX(3615{,}257) = 0x0E1F$$

Now both cell 0x04 and cell 0x0F must be changed:
1. Cell 0x04 to be changed from 0xFFFF (old value) to 0xAE0D (new value)
2. Cell 0x0F to be changed from 0x099A (old value) to 0x0E1F (new value)

6 Uploading the new values in the MLX90614 EEPROM

For more detail regarding the SMBus communication refer to *SMBus communication with MLX90614*

6.1 Procedure for MLX90614xAx and MLX90614xBx type devices

6.1.1 Write 0x0000 to address 0x04 (erase the EEPROM cell)

Opcode	Command	Number of bytes (bits)*	Layout
0010 0100	EEPROM Access	5 bytes (47 bits)	SA + Command + 0x0000 + PEC

Note: The number of bits includes all bits (e.g. start, repeated start, stop and acknowledge bits)*

6.1.2 Write the new value to address 0x04

Opcode	Command	Number of bytes (bits)*	Layout
0010 0100	EEPROM Access	5 bytes (47 bits)	SA + Command + 2 bytes of data + PEC

Note: The number of bits includes all bits (e.g. start, repeated start, stop and acknowledge bits)*

6.1.3 Read the value in address 0x04 in order to check that the correct value is stored

Opcode	Command	Number of bytes (bits)**	Layout
0010 0100	EEPROM Access	6 bytes (57 bits)	SA + Command + SA + 2 bytes of data + PEC

Note: The number of bits includes all bits (e.g. start, repeated start, stop and acknowledge bits)*

6.1.4 Restart the module

6.2 Procedure for MLX90614xCx type devices

6.2.1 Enter EEPROM address 0x0F unlock key

Opcode	Command	Number of bytes (bits)*	Layout
0110 0000	Enter EEPROM address 0x0F unlock key	3 bytes (29 bits)	SA + Command + PEC

Note: The number of bits includes all bits (e.g. start, repeated start, stop and acknowledge bits)*

6.2.2 Write 0x0000 to address 0x04 (erase the EEPROM cell)

Opcode	Command	Number of bytes (bits)*	Layout
0010 0100	EEPROM Access	5 bytes (47 bits)	SA + Command + 0x0000 + PEC

Note: The number of bits includes all bits (e.g. start, repeated start, stop and acknowledge bits)*

6.2.3 Write the new value to address 0x04

Opcode	Command	Number of bytes (bits)*	Layout
0010 0100	EEPROM Access	5 bytes (47 bits)	SA + Command + 2 bytes of data + PEC

Note: The number of bits includes all bits (e.g. start, repeated start, stop and acknowledge bits)*

6.2.4 Read the value in address 0x04 in order to check that the correct value is stored

Opcode	Command	Number of bytes (bits)**	Layout
0010 0100	EEPROM Access	6 bytes (57 bits)	SA + Command + SA + 2 bytes of data + PEC

Note: The number of bits includes all bits (e.g. start, repeated start, stop and acknowledge bits)*

6.2.5 Write 0x0000 to address 0x0F (erase the EEPROM cell)

Opcode	Command	Number of bytes (bits)*	Layout
0010 1111	EEPROM Access	5 bytes (47 bits)	SA + Command + 0x0000 + PEC

Note: The number of bits includes all bits (e.g. start, repeated start, stop and acknowledge bits)*

6.2.6 Write the new value to address 0x0F

Opcode	Command	Number of bytes (bits)*	Layout
0010 1111	EEPROM Access	5 bytes (47 bits)	SA + Command + 2 bytes of data + PEC

Note: The number of bits includes all bits (e.g. start, repeated start, stop and acknowledge bits)*

6.2.7 Read the value in address 0x0F in order to check that the correct value is stored

Opcode	Command	Number of bytes (bits)**	Layout
0010 1111	EEPROM Access	6 bytes (57 bits)	SA + Command + SA + 2 bytes of data + PEC

Note: The number of bits includes all bits (e.g. start, repeated start, stop and acknowledge bits)*

6.2.8 Restart the module